Interoperability between Digital Earth and DODS

C. Lynnes, G. Serafino, P. Sweatman, R. Mullinix, N. Pollack
NASA Goddard Space Flight Center
Code 902, Greenbelt, MD, 20771
P. Cornillon, D. Holloway, J. Gallagher
Data Access Systems, LLC
P.O.Box 6, Saunderstown, RI 02874, U.S.A

Abstract- **The DODS-WMT Gateway server is a Web Map Server (WMS) utilizing the Distributed Oceanographic Data System (DODS) data access protocol (DAP) to access data for the visualization and formatting operations required to satisfy OpenGIS Consortium (OGC) Web Mapping Testbed (WMT) map requests. The Gateway design partitions the WMS operations into a number of packages to handle the primary activities required for compliance as an OGC WMT-1.0 Web Map Server. These activities include handling the client request parameters (MapRequest), advertising the Map Server's capabilities (Capabilities), and producing the various Map products requested by the remote client (Map). The DODS-WMT Gateway design utilizes interfaces to external 'plug-in' visualization, formatting, projection, and catalog specializations, which can be readily customized for specific data products.**

I. Introduction

The DODS-WMT Gateway project grew out of the serendipitous similarities in functionality and protocol between the Distributed Oceanographic Data System (DODS)[1] and the Web Mapping Testbed (WMT) of the OpenGIS Consortium[2]. The two systems were initially developed quite separately. The DODS began as an attempt to provide oceanographic data stored in a variety of formats on distributed servers. On the other hand, WMT was targeted at Geographic Information Systems (GIS) users and adopted as the primary technology underlying the Digital Earth program. However, both systems standardize data transport to allow users to use their favorite analysis clients; both support spatial subsetting at the server; and both use the Hypertext Transfer Protocol (HTTP) and overloaded Universal Resource Locators (URLs) to implement the request protocol. It thus seemed feasible to adapt DODS servers to service WMT requests. This would allow data centers to maintain enhanced DODS servers to satisfy both DODS and WMT requests, thus leveraging the format handling (Hierarchical Data Format, network Common Data Format and flat binary) and subsetting capabilities inherent in DODS. This is of particular interest to the Goddard Distributed Active Archive Center (DAAC), a component of NASA's Earth Observing System which has a number of datasets already available through DODS with more on the way.

A. DODS-WMT Gateway Components

Fig. 1 shows the DODS-WMT Gateway, its components, and how those interact with other software when the server is asked to handle a request. A WMT client sends a request using HTTP to the DODS-WMT Gateway server. This request is initially handled by the server's HTTP daemon (httpd), which uses the CGI 1.1 interface to pass parameters to the DODS-WMT Gateway. Once the DODS-WMT Gateway receives the request parameters it can begin to process the request. The Gateway obtains the data from a DODS server by issuing a DODS Data Access Protocol (DAP) request. The Gateway then converts the data into a mapped visualization matching the projection, bounds, image size and format requested by the client (Fig. 2).

Revision 1.0.0 of the WMT interface implementation specification[2] defined a set of request types that a WMT compliant server must support. These are defined as the GetCapabilities, GetMap, and GetFeatureInfo request types. As most of the DAAC's DODS-served datasets are numerical grids rather than vector features, only the GetCapabilities and GetMap features are as yet implemented in the DODS-WMT Gateway.

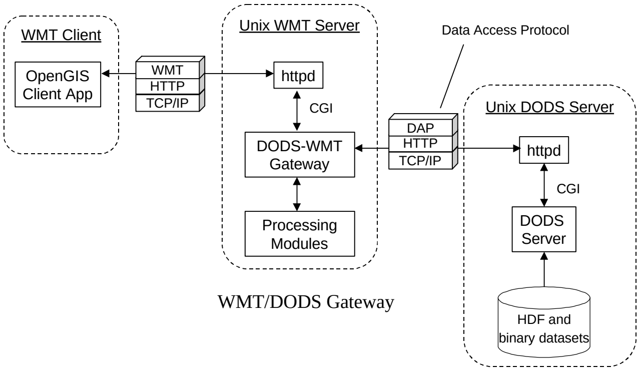

Fig. 1. Architecture of the DODS-WMT Gateway.

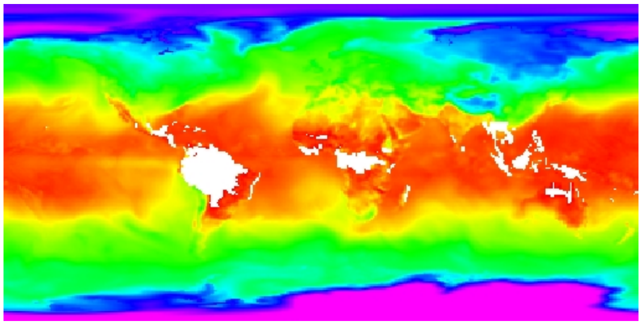

Fig. 2. Map visualization generated by DODS-WMT Gateway for global ground temperature, skin temperature of the surface (sea surface temperature over water) from the Data Assimilation System's 2-D late-look synoptic assimilation.

1) GetCapabilities. In response to a GetCapabilities request, the DODS-WMT Gateway returns a Capabilities eXtensible Markup Language (XML) document read from the local host's file-system. The Capabilities XML document reflects the current state of advertised capabilities for a particular DODS-WMT Gateway server. The information provided in the server's Capability XML document range from its supported version, request and format types, to specific data layers, their styles or visualization provided by the server, as well as the spatial reference system information specific to each data layer. Additionally, the Capabilities XML document can provide supplementary information regarding the server, as well as for the individual data layers, either directly in the document itself or through external references using the MetadataURL elements.

2) GetMap. With a GetMap request, the WMT client requests a rendered map (picture) precisely matching the desired geographic projection, picture size, and style. When data are stored as integer or floating point numbers, rather than as a raster image, the DODS-WMT gateway must subset, interpolate and render a map corresponding precisely to the requests. In response to a GetMap request, the DODS-WMT Gateway translates selected client request parameters and data layer information contained in the server's Capabilities XML document into a DODS data request. The data retrieved from the DODS server are provided to data visualization and formatting objects, which operate to construct the requested map product.

The majority of science datasets targeted for serving via this Gateway are multi-file in nature, with the granularity typically defined by time. Revision 1.0.0 of the WMT specification[2] did not address selecting single granules from datasets comprised of multiple granules spread out in space and/or time. Initially, the DODS-WMT Gateway used WMT Vendor Specific Parameter (VSP) elements for selecting files by time. Revision 1.0.3 of the WMT XML DTD specification provided a formal mechanism for representing multiple granules in space and/or time through the Dimension and Extent elements. The DODS-WMT Gateway was modified to support these new features of the Capabilities XML specification, resulting in a substantive design change to supporting specializations.

III. Classes in the DODS-WMT Gateway

The DODS-WMT Gateway class uses several primary classes to respond to a WMT client request. These classes are used to represent the server's capabilities, the client's request parameters, and provide mechanisms for accessing DODS-served data and transforming it into the requested WMT-compliant Map product.

A. DODS-WMT Gateway

The primary DODS-WMT Gateway server module is not implemented formally as a C++ class. However, it is simplest to think of it as a C++ class, which has a main method. This software, whether actually a collection of functions or a formal class, instantiates the other classes shown in Fig. 3 in response to a request.

The DODS-WMT Gateway server module interrogates the request parameters provided by the server's httpd daemon, and instantiates the objects responsible for satisfying the request type. The valid request types for WMT 1.0.0 are Capabilities and Map. (A Coverage response has also been prototyped.)

B. Managing Request Parameters: MapRequest Class

Client requests may be presented using either HTTP's GET or POST mechanism. The MapRequest object parses the parameters and saves the content. It also provides methods that compare the client's request parameters against the server's advertised capabilities to insure that the client's request can be satisfied. If the server cannot satisfy the request, the MapRequest object instructs the server to return an exception to the client. This object is continually referred to during the construction of the map products, providing access to the client request parameters to the various objects satisfying the request.

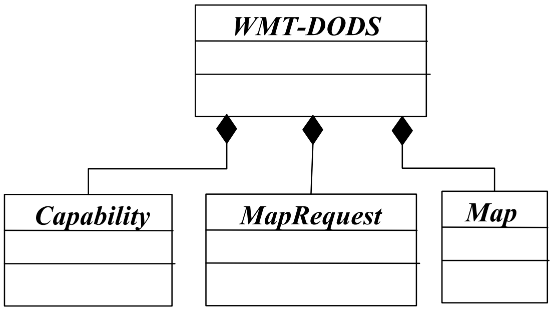

Fig. 3. WMT-DODS object diagram.

C. *Managing Advertised Services: Capabilities Class*

The DODS-WMT Gateway responds to a Capabilities request by returning an XML document to the client. This is currently implemented by reading a static file. However, rather than have the DODS-WMT Gateway object itself handle reading the XML document, the Capabilities class hides the actual process used to generate the document so that we can easily extend its functionality to dynamic generation in future revisions.

The Capabilities object is a repository for the server's advertised services and data layers described in the server's Capabilities XML document. This functionality is used by DODS/WMT objects in two key ways; verifying client requests against the server's advertised capabilities, and mapping client request parameters to specific request type and data layer information.

D. *Map Class*

The Map object (Fig. 4) is responsible for satisfying WMT client map requests. It retrieves request parameters from the MapRequest object and the server's advertised capabilities from the Capabilities object, and based on those parameters, produces a WMT-compliant Map product, which is returned to the client.

Unless an error is detected and an Exception is returned, the DODS-WMT Gateway generates an HTTP response containing the requested Map product encapsulated within the HTTP MIME header. If an error is detected, the DODS-WMT Gateway returns an error using XML if so defined given the Exceptions parameter. If the Map object detects the error and generates a message returned in a map format, the DODS-WMT Gateway returns the Exception in that format.

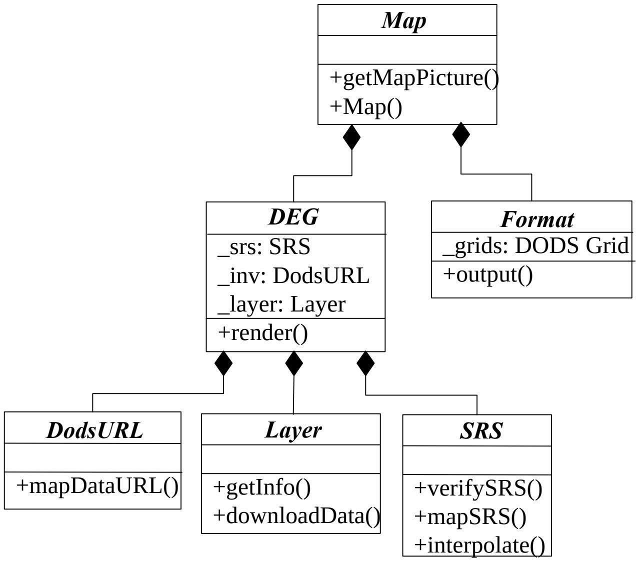

Fig. 4. Object diagram for Map class.

One of the primary roles of the Map object is to identify the style, Spatial Reference System (SRS), and format specializations (e.g., Portable Network Graphics or Joint Photographic Experts Group) required to satisfy the client map request, and instantiate the proper specializations at run-time for these elements. In order to satisfy the arbitrary number of potential layer-style combinations available for science data, as well as SRS and output formats, the DODS-WMT Gateway utilizes a number of abstract base classes to define standard interfaces through which these specializations are implemented. The current set of abstract base classes employed by the Gateway includes Display Element Generator (DEG), SRS, and Format. The DodsURL class is used as a base class but is not an abstract base class.

A Map object may contain multiple instances of specialized DEG objects. This accommodates the WMT specification's requirement that a WMT server support requests that combine, or overlay, several layer/style instances. Each instance of a DEG serves as a (smart) container for a requested layer/style map product.

1) DEG (Display Element Generator). The DEG base class defines a standard interface through which all visualization, or WMS 'style' operations performed by the Gateway must operate. The primary interface used by the Map and Coverage classes to interface with the DEG specializations is named 'render'.

The DEG class, and all specializations based on it, is one of the primary components of the DODS-WMT Gateway design. Each individual layer/style contained in the map request generates a separate, specialized DEG object to produce the visualization, or map style requested. This well-defined interface allows the server to 'plug-in', or customize the visualization (style) for different data sets. Clearly these customizations can be added by adding C++ derived classes. However, we have also add a C++ specialization to the DODS/WMT Gateway that calls external executables to interpolate and render images. This allows the addition of datasets and their varied rendering and interpolation methods without recompiling the DODS-WMT Gateway software.

Each separate DEG object maintains a set of Layer objects consistent with the client's map request. Because the WMT specification obscures the multi-granular nature of the underlying science data the DEG object must translate the client map request into one, or many individual Layer objects which will contain the actual data used to generate the requested map.

Each separate DEG object also maintains a specialized SRS object that is instantiated based on the client's SRS request parameters, and consistent with the advertised SRS information contained in the server's Capability XML document for the particular layer/style requested.

2) SRS (Spatial Reference System). The SRS base class defines a standard interface for all Spatial Reference System (SRS) mapping operations performed by the Gateway. The primary task of the SRS object is mapping the client's SRS and bounding box (BBOX) request parameters into a DODS constraint expression (CE), which is used to extract the subset of data needed to satisfy the map request. To accomplish this, the SRS object interrogates the remote DODS data set, and using information contained in the server's Capability XML document, maps the BBOX request parameters into a CE for the particular data set, or layer requested. The Layer objects use the resulting CE to form the complete the URL to request information from a DODS data server. Because the SRS object maps the WMS BBOX to a DODS CE, the DODS-WMT Gateway can make full use of the DODS' constraint-based access features to limit the amount of data read and transferred to only that information requested.

The SRS 'interpolate' method is responsible for taking a DODS Grid variable and interpolating it to match the SRS and BBOX request parameters. Note that this involves a resampling of the data; thus the resulting map is a visual representation of the data, not the actual data.

3) Layer. The Layer object is the repository for the actual science data used to make the requested map product. It is an abstraction of the layer concept described by the WMT specification. Revision 1.0.0 of the WMT specification used the term layer to define a singular data product, but this obscures the potential multi-granular nature of the underlying science data. To address this, the DODS-WMT Gateway design defines Layer objects to contain the individual granules comprising a client request. To conform to the WMT layer concept, the DEG object employs a vector of Layer objects to contain the multiple DODS data granules referenced by the client request as a single entity.

The Layer object is the sole mechanism to interact with the DODS servers, providing the science data served by the DODS-WMT Gateway. It provides methods for retrieving data from DODS servers as well as methods used by the SRS object for querying the DODS served data to map BBOX request parameters into a DODS constraint expression. The Layer object constructs a fully qualified URL to a DODS server, using a base URL provided by the DEG object and a CE provided by the SRS object. The retrieved science data is held by the Layer object in an instance of either the DODS' Grid or Array class. The Layer object also provides methods used to access these DODS data objects.

4) DodsURL. The DodsURL base class is used to map a Layer name in the server's Capabilities XML document to a URL to a DODS-served data set. The Capabilities XML document is intended to provide sufficient information for a client to request a map product but does not offer much flexibility for storing server specific catalog information, especially for multiple-file and multi-parameter datasets. The DODS-WMT Gateway requires a mechanism to facilitate access to catalog information typically used to provide base URLs to DODS served data sets.

The initial DodsURL implementation used the DataURL element within the Layer element for the requested layer/style advertised in the server's Capabilities XML document. For single file data sets, the DataURL element alone was sufficient to provide a base URL to a DODS served data set.

For data sets comprised of multiple files, the DataURL element alone is insufficient to qualify the potential set of base URLs contained in a given client request. To satisfy this condition, a CatalogURL class was designed using DodsURL as a base class. The CatalogURL object required the use of Vendor Specific Parameters (VSPs) that a client uses to request various files from the data layer. The problem with this approach is twofold. Firstly, VSPs are not easily understood by WMT clients, making it difficult to develop WMT client applications. Secondly, a single catalog specialization based on DodsURL cannot be continually extended to handle additional cataloging problems without becoming unmanageable.

Eventually, the DODS-WMT Gateway extended the initial DodsURL, and CatalogURL implementations into a set of catalog specializations, similar to those used for DEG, SRS, and Format operations. The ability to customize the catalog interface between the server's Capabilities XML representation, and the underlying dataset itself, can facilitate making data available to the WMT community via the Gateway. Additionally, customized catalog specializations permit the DEG objects to reference multiple variables within a dataset, and/or multiple datasets in response to a single layer/style advertised in the Gateway's Capabilities XML document. This implies that complex layer/style combinations can be supported, aggregating or fusing data from multiple sources into an advertised map product.

To support these features, the catalog specializations use the Keywords element within the individual Layer elements advertised in the server's Capabilities XML document to store this information. This syntax allows only one catalog specialization per advertised layer/style element in the Capabilities XML document, but the catalog object itself may allow an arbitrary number of datasets, and/or data variables, to be referenced for the given layer/style.

While the WMT specification allows a great deal of flexibility in the use of the Keywords element, and several other elements, in the Capabilities XML document, this use of the Keywords element is problematic. This information could be stored in a separate file from the Capabilities XML document, but at the risk of manageability. In any case, as the OGC Stateless Catalog or Registry service specification becomes available, it's possible this approach will need to be redesigned to comply with that specification.

5) Format. The Format base class defines a standard interface through which all WMS output format operations performed by the Gateway must operate. The primary interface used by the Map and Coverage classes to interface with the Format specializations is named 'output'.

Format specialization classes can be defined to support the wide variety of WMS output types. The initial implementation of the DODS-WMT Gateway supported GIF and GeoTIFF for map requests, and GeoTIFF and HDF-EOS for coverage requests.

Each Format specialization's output method is responsible for taking the DODS Grid objects provided by the DEG objects, and producing the map product in the requested output format, subject to the Height, Width, Transparent, and BGColor parameters of the Map request. See sections [6.2.8.2.4-7 of the WMT specification].

The specialized Format objects write the output directly to standard output, which the Format object assumes is the httpd's socket connected back to the client process.

IV. Conclusion

The prototype DODS-WMT Gateway has demonstrated the feasibility and usefulness of using DODS to facilitate implementation of WMT services. While specializations must be added to convert DODS-served numerical datasets into suitable rasterized responses to GetMap requests, the DODS-WMT gateway provides a framework for these specializations. Over time, a library of these specializations will accumulate so that eventually, adding a new dataset can be completely driven by configuration files. The development of a generalized gateway (which can accept specializations) also identified a number of challenges to be overcome in making NASA remote sensing data available via the WMT, specifically the handling of the temporal dimension, many-layered datasets, and point time series.

The implementation of this prototype as a Gateway, rather than as the originally envisioned server enhancements, means that the gateway need not be co-located with the DODS server. Following this line of reasoning, a DODS-WMT Gateway could be extended to use several distributed DODS servers on the back end. This bears some similarity to the cascading map server concept in OpenGIS.

Acknowledgments

We thank the Earth Science Technology Office at NASA for funding the development of WMT/DODS.

References

[1] J. Gallagher and G. Milkowski, 1995. Data Transport Within The Distributed Oceanographic Data System, Fourth International World Wide Web Conference, December 11-14, Boston, Massachusetts, USA, http://www.w3.org/Conferences/WWW4/Papers/67/

[2] OpenGIS Consortium, 2000. "OpenGIS® Web Map Server Interfaces Implementation Specification, Version 1.0.0", http://www.opengis.org/techno/specs/00-028.pdf.